UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/738,371	12/16/2003	Steven Soltis	3000	7266

7590 06/10/2008

Beck & Tysver, P.L.L.C.
Suite 100
2900 Thomas Avenue S.
Minneapolis, MN 55416

EXAMINER	
KANG, PAUL H	
ART UNIT	PAPER NUMBER
2144	
MAIL DATE	DELIVERY MODE
06/10/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/738,371	SOLTIS, STEVEN	
	Examiner	Art Unit	
	Paul H. Kang	2144	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 August 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-56* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-56* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *16 December 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *8/21/06*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Double Patenting

1. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

2. Claims 1-43 of US Pat. No. 6,697,846 B1contain every element of claims 1-56 of the instant application and as such anticipate claims 1-56 of the instant application.

"A later patent claim is not patentably distinct from an earlier patent claim if the later claim is obvious over, or **anticipated by**, the earlier claim. In re Longi, 759 F.2d at 896, 225 USPQ at 651 (affirming a holding of obviousness-type double patenting because the claims at issue were obvious over claims in four prior art patents); In re Berg, 140 F.3d at 1437, 46 USPQ2d at 1233 (Fed. Cir. 1998) (affirming a holding of obviousness-type double patenting

where a patent application claim to a genus is anticipated by a patent claim to a species within that genus). " ELI LILLY AND COMPANY v BARR LABORATORIES, INC., United States Court of Appeals for the Federal Circuit, ON PETITION FOR REHEARING EN BANC (DECIDED: May 30, 2001).

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 1-56 are rejected under 35 U.S.C. 102(b) as being anticipated by T. Anderson, et al., "Serverless Network File Systems," Proceedings of the Fifteenth ACM Symposium on Operating System Principles, 1995 (hereinafter referred to as Anderson).

5. As to claim 1, Anderson teaches a shared storage distributed file system having a namespace defining a directory structure of files and metadata that includes pointers to real-data, the file system comprising:

a) a network storage device and at least one server computer running server software for managing the namespace (in the distributed system, multiple metadata mangers share load, sections 1 and 3.1; See also Fig. 2); and

b) a plurality of client computers each running client software, the client software

i) issuing namespace requests to the server computer (section 3.1), and

ii) directly retrieving, analyzing, and altering the metadata (configuration in which all nodes include metadata managers, metadata is directly modified, Fig. 2; sections 1 and 3.1).

6. As to claim 2, Anderson teaches the file system wherein metadata includes allocation tables that store information identifying data as allocated and not allocated (sections 3.1.1-3.1.4).

7. As to claim 3, Anderson teaches the file system wherein the client software directly generates metadata pointers to real-data (sections 1 and 3.1).

8. As to claim 4, Anderson teaches the file system wherein the server software enforces file access permissions (Section 3.2.3).

9. As to claim 5, Anderson teaches the file system wherein the server software manages the namespace in response to namespace requests from the client computers, including requests to add new filenames to the namespace, to remove existing filenames from the namespace, and to search the namespace for filenames (sections 3.1 and 3.1.1-3.1.3).

10. As to claim 6, Anderson teaches the file system wherein the namespace search for filenames returns information necessary to retrieve the metadata (sections 3.1 and 3.1.1-3.1.3).

11. As to claim 7, Anderson teaches the file system wherein the server software enforces file permissions during the namespace search (Section 3.2.3).

12. As to claim 8, Anderson teaches the file system wherein the client computers directly respond to file system requests concerning a file from an application program (Fig. 2; sections 1, 3.1, and 3.2.2).

13. As to claim 9, Anderson teaches a network of connected computing devices for implementing a shared storage distributed file system, the file system having a namespace, real-data, and metadata, the network comprising:

a) a network storage device connected to a network (Fig. 2);

b) a server computer that manages the namespace in response to namespace requests, including requests to add new filenames to the namespace and to remove existing filenames from the namespace (in the distributed system, multiple metadata mangers share load, sections 1 and 3.1; See also Fig. 2); and

c) a client computer in network communication with the server computer and the network storage device, wherein the client computer

i) issues namespace requests to the server computer (section 3.1),

ii) reads and writes the real-data directly from the network storage device (sections 1, 3.1, 3.1.1 and 3.2.2), and

iii) creates, retrieves, and modifies the metadata (configuration in which all nodes include metadata managers, metadata is directly modified, Fig. 2; sections 1, 3.1, and 3.2.2).

14. As to claim 10, Anderson teaches the network wherein the client computer communicates with the server computer via a local area network, and the client computer communicates with the network storage device via a storage area network (Fig. 2; sections 1, and 3.1).

15. As to claim 11, Anderson teaches the network wherein namespace requests are communicated via the local area network (Fig. 2; sections 1, and 3.1).

16. As to claim 12, Anderson teaches the network wherein the client reads and writes the real-data via the storage area network (Fig. 2; sections 1, 3.1, 3.1.1-3.1.3, and 3.2.2-3.2.3).

17. As to claim 13, Anderson teaches the network wherein the client computer requests file attributes from the server computer (Fig. 2; sections 1, and 3.1).

18. As to claim 14, Anderson teaches the network wherein file attributes are communicated via the local area network (Fig. 2; sections 1, and 3.1).

19. As to claim 15, Anderson teaches the network wherein the client computer acquires a lock prior to modifying the metadata (Section 3.2.3).

20. As to claim 16, Anderson teaches the network wherein the namespace requests include requests to search the namespace for filenames (Fig. 2; sections 1, 3.1 and 3.1.1-3.1.3).

21. As to claim 17, Anderson teaches the network wherein the server computer enforces file access permissions during the namespace search (Section 3.2.3).

22. As to claim 18, Anderson teaches the network wherein the client computer stores the metadata by requesting that the metadata be stored by the server computer (Fig. 2; sections 1, 3.1, and 3.1.1-3.1.3).

23. As to claim 19, Anderson teaches the network wherein the server computer stores the metadata within a real-data file (sections 3.1-3.1.3 and 3.2.2).

24. As to claim 20, Anderson teaches the network wherein the server computer is operating as the server of a client-server file system to store the metadata (sections 3.1-3.1.3).

25. As to claim 21, Anderson teaches the network wherein the server computer stores the metadata on a server storage device locally attached to the server computer (Fig. 2; sections 1, 3.1 and 3.1.1-3.1.3).

26. As to claim 22, Anderson teaches the network wherein the server computer stores the metadata on a server storage device different from the network storage device (Fig. 2; sections 1, 3.1 and 3.1.1-3.1.3).

27. As to claim 23, Anderson teaches an improved file system comprising:

a) a storage device (Fig. 2);

b) a server software program that runs on a server computer and maintains a namespace (in the distributed system, multiple metadata mangers share load, sections 1 and 3.1; See also Fig. 2); and

c) a client software program that runs on a client computer that responds to file system requests from an application program concerning a file, wherein the client software

i) obtains addressing metadata containing at least one pointer addressing real-data for the file (section 3.1),

ii) uses the addressing metadata to locate real-data associated with the file on the storage device (sections 1, 3.1 and 3.1.1), and

iii) alters the addressing metadata for the file (configuration in which all nodes include metadata managers, metadata is directly modified, Fig. 2; sections 1, 3.1, and 3.2.2).

28. As to claim 24, Anderson teach the improved file system wherein the server software adds new filenames to the namespace, removes existing filenames from the namespace, and searches the namespace for filenames (Fig. 2; sections 1, 3.1 and 3.1.1-3.1.3).

29. As to claim 25, Anderson teach the improved file system wherein the client software sends a namespace search request to the server software in order to obtain addressing metadata (Fig. 2; sections 1, 3.1 and 3.1.1-3.1.3).

30. As to claim 26, Anderson teach the improved file system wherein the server software enforces file access permissions during the namespace search (Section 3.2.3).

31. As to claim 27, Anderson teach the improved file system wherein the addressing metadata is found within an inode obtained by the client software (Sections 3.1.1-3.1.3).

32. As to claim 28, Anderson teach the improved file system wherein the client software further obtains allocation table metadata concerning allocation of storage on the storage device and modifies the allocation table metadata when performing file allocation and de-allocation (Sections 3.1.1-3.1.4).

33. As to claim 29, Anderson teach the improved file system wherein the allocation table metadata is a bitmap table (Sections 3.1.1-3.1.4).

34. As to claim 30, Anderson teach the improved file system wherein the addressing metadata is found within an inode obtained by the client software (Sections 3.1.1-3.1.4).

35. As to claim 31, Anderson teaches an improved file system comprising:

a) a storage device containing real-data and one or more direct pointers addressing the real-data (Fig. 2);

b) a server software program that runs on a server computer, the server software program

i) maintains a namespace, and ii) stores an indirect pointer within the namespace related to a file, the indirect pointer addressing at least one file related direct pointer on the storage device (in the distributed system, multiple metadata mangers share load, sections 1 and 3.1; See also Fig. 2); and

c) a client software program that runs on a client computer that responds to file system requests from an application program concerning the file; the client software program

i) obtains the indirect pointer for the file from the server software program (section 3.1),

ii) uses the indirect pointer to obtain the file related direct pointer directly from the storage device (Fig. 2; sections 1 and 3.1), and

iii) uses the file related direct pointer to read and write real-data associated with the file directly from the storage device (configuration in which all nodes include metadata managers, metadata is directly modified, Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.2).

36. As to claim 32, Anderson teaches the improved file system wherein the client software program modifies the file related direct pointer during file write operations (Section 3.2.2).

37. As to claim 33, Anderson teaches the improved file system wherein the client software acquires a lock prior to modifying the file related direct pointer (Section 3.2.3).

38. As to claim 34, Anderson teaches the improved file system wherein the server software services namespace requests from the client computer, including requests to add new filenames to the namespace, to remove existing filenames from the namespace, and to search the namespace for filenames (sections 1, 3.1, 3.1.1-3.1.4, and 3.2.1-3.2.2).

39. As to claim 35, Anderson teaches the improved file system wherein the client software further obtains allocation table metadata concerning allocation of storage on the storage device and updates the allocation metadata when performing file allocation and de-allocation (sections 1, 3.1, 3.1.1-3.1.4, and 3.2.1-3.2.2).

40. As to claim 36, Anderson teaches the improved file system wherein the client software acquires a lock prior to updating the allocation table metadata (Section 3.2.3).

41. As to claim 37, Anderson teaches an improved file system comprising:

a) a storage device (Fig. 2);

b) a server computer running server software that maintains a namespace defining a directory structure of files (sections 1 and 3.1; See also Fig. 2), and

c) a client computer in network communication with the server computer and the storage device, the client computer running client software that

i) obtains allocation information concerning the allocation of storage on the storage device, and ii) alters the allocation information during file allocation and de-

allocation (configuration in which all nodes include metadata managers, metadata is directly modified, Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.2).

42. As to claim 38, Anderson teaches the improved file system of claim 37 wherein the server software accesses and modifies the directory structure in response to namespace requests from the client computer, including requests to add new filenames to the namespace, to remove existing filenames from the namespace, and to search the namespace for filenames (Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.1-3.2.2).

43. As to claim 39, Anderson teaches the improved file system wherein the client software acquires a lock prior to obtaining the allocation information (Section 3.2.3).

44. As to claim 40, Anderson teaches the improved file system wherein the client software further iii) obtains addressing metadata locating real-data for a particular file; iv) uses the addressing metadata to access real-data associated with the particular file on the storage device, and v) alters the addressing metadata for the file (Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.1-3.2.2).

45. As to claim 41, Anderson teaches the improved file system wherein the client software acquires a lock prior to altering the addressing metadata (Section 3.2.3).

46. As to claim 42, Anderson teaches the improved file system wherein the allocation information is obtained from the storage device and the altered allocation information is stored on the storage device (sections 1, 3.1, 3.1.1, and 3.2.2).

47. As to claim 43, Anderson teaches a network of connected computer devices comprising:

a) a first computer running software for

i) managing a directory structure of files, and ii) servicing directory requests, the directory requests including requests to add filenames to the directory, remove filenames from the directory, and search the directory (in the distributed system, multiple metadata mangers share load, sections 1 and 3.1; See also Fig. 2); and

b) a second computer running software for

i) submitting to the first computer directory requests relating to a file request, and ii) analyzing and altering metadata relating to the file request, the metadata including pointers to real-data (configuration in which all nodes include metadata managers, metadata is directly modified, Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.2).

48. As to claim 44, Anderson teaches the network wherein the metadata includes data allocation information (sections 1, 3.1, 3.1.1, and 3.2.2).

49. As to claim 45, Anderson teaches the network wherein the second computer directly responds to file requests from an application program (sections 1, 3.1, 3.1.1, and 3.2.2).

50. As to claims 46-49, Anderson teaches the network wherein the first computer enforces file access permissions for requests received from the second computer, while adding a filename to a directory, while removing a filename from a directory, and while searching for a filename within a directory (Section 3.2.3).

51. As to claims 50, 52, and 54, Anderson teaches a method for handling a file request from an application, the file request relating to real-data of a file, the real-data being stored on a network connected, shared storage device, the method comprising:

a) receiving the file request from the application at a client computer; b) requesting an indirect extent pointer for the file from a server computer; c) receiving the requested indirect extent pointer at the client computer; d) using the indirect extent pointer to retrieve metadata from the storage device (sections 1, 3.1, 3.1.1);

e) for a file read request, the client computer i) analyzing the metadata to determine the locations of the real-data stored on the storage device, and ii) reading the real-data from the storage device (Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.1); and

f) for a file write request, the client computer i) analyzing the metadata to determine the locations of the real-data stored on the storage device, ii) allocating additional storage space to the file, iii) writing real-data to the storage device, iv) updating metadata, and v) storing updated metadata on the storage device (Fig. 2; sections 1, 3.1, 3.1.1, and 3.2.2).

52. As to claim 51, Anderson teaches the method wherein during the allocation of additional storage space for the file write request, the client computer retrieves, analyses, and modifies the allocation table metadata (sections 1, 3.1, 3.1.1, and 3.2.2).

53. As to claim 53, Anderson teaches the method wherein a lock is acquired prior to altering the metadata relating to the location of real-data (Section 3.2.3).

54. As to claim 55, Anderson teaches the method wherein the allocation information consists of bitmap tables (Sections 3.1.1-3.1.4).

55. As to claim 56, Anderson teaches the method wherein a lock is acquired prior to analyzing and altering allocation information metadata, and the lock is released after saving the allocation information (Section 3.2.3).

Conclusion

56. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Paul H. Kang whose telephone number is (571) 272-3882. The examiner can normally be reached on IFP.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Vaughn can be reached on (571) 272-3922. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Paul H Kang/
Primary Examiner
Art Unit 2144